Yale University
Department of Computer Science

When Do Extra Majority Gates Help?

Polylog(n) Majority Gates are Equivalent to One

Richard Beigel

YALEU/DCS/TR-885
December 20, 1991

When Do Extra Majority Gates Help?

Polylog(n) Majority Gates are Equivalent to One

Richard Beigel*

Yale University

Abstract

Suppose that f is computed by a constant depth circuit with 2^m AND-, OR-, and NOT-gates, and m majority-gates. We prove that f is computed by a constant depth circuit with $2^{m^{O(1)}}$ AND-, OR-, and NOT-gates, and a single majority-gate, which is at the root.

One consequence is that if f is computed by an AC^0 circuit plus polylog majority-gates, then f is computed by a probabilistic perceptron having polylog order. Another consequence is that if f agrees with the parity function on three-fourths of all inputs, then f cannot be computed by a constant depth circuit with $2^{n^{o(1)}}$ AND-, OR-, and NOT-gates, and $n^{o(1)}$ majority-gates.

1. Introduction

One of the goals of complexity theory is to find ways to reduce the use of one resource. Typically this entails a modest increase in some other resources.

Recently, quasipolynomial size circuits have been the setting for unexpected resource tradeoffs [1, 2, 16, 7, 14, 5]. In this paper we show how to reduce the number of majority-gates in many kinds of quasipolynomial size circuits from polylog down to 1.

For example, consider constant-depth quasipolynomial-size circuits that consist of AND-, OR-, NOT-, and majority-gates. We show how to reduce the number of majority-gates from m to 1, while increasing the number of other gates by $2^{m \operatorname{polylog} n}$ and increasing the depth by 2. As a corollary we convert such circuits to perceptrons having small order. As another corollary, we unify several known lower bounds [4, 15, 9, 13] for computing parity with constant depth circuits, proving that if a bounded depth circuit consisting of AND-, OR-, NOT-, and majority-gates computes parity correctly on three-fourths of all inputs, then the circuit must have exponential size or a polynomial number of majority-gates.

*Dept. of Computer Science, 51 Prospect Street, P.O. Box 2158, Yale Station, New Haven, CT 06520-2158. Email: beigel-richard@cs.yale.edu. Supported in part by NSF grant CCR-8958528.

Our result is an exponential improvement on the bounds in [6]. The proof uses the low-degree polynomials developed in that paper and an observation of Fortnow and Reingold [8] in a nontrivial way.

We also show how to reduce the number of symmetric gates in many kinds of quasipolynomial size circuits from $O(\log\log n)$ down to 1. Consider constant-depth quasipolynomial size circuits that consist of AND-, OR-, NOT-, and MOD_m-gates and symmetric gates. We show how to reduce the number of symmetric gates from m to 1, while increasing the number of other gates by $2^{2^m \operatorname{polylog} n}$ and increasing the depth by 1. The proof uses base-B representation as in [11]. As a corollary we extend results of [16, 7] on ACC, showing that any function computed by a constant-depth, quasipolynomial-size circuit consisting of AND-, OR-, NOT-, and MOD_m-gates and $O(\log\log n)$ symmetric gates is in fact computed by a depth 2, quasipolynomial size circuit with a symmetric gate at the root and AND-gates having polylog fanin at the bottom level.

Our results for decision problems contrast with those of Amir et al [3] who showed in broad generality that extra oracle gates do help to compute additional *functions.*

2. Representing Boolean functions

Boolean values are often represented as elements of $\{0,1\}$, 0 denoting false, and 1 denoting true. They can also be represented as elements of $\{-1,1\}$, -1 denoting false and 1 denoting true. Real polynomials or rational functions in either representation can be converted to the other without affecting the degree.

A majority-gate outputs true if more than half of the inputs are true. By standard techniques, we will assume that it is never the case that exactly half of the inputs to a majority-gate are true.

In the first representation, the majority function is

$$\mathrm{MAJ}(x_1,\ldots,x_n) = \begin{cases} 1 & \text{if } \sum x_i > \frac{1}{2}n, \\ 0 & \text{if } \sum x_i < \frac{1}{2}n. \end{cases}$$

In the second representation,

$$\mathrm{MAJ}(x_1,\ldots,x_n) = \mathrm{sgn}(\textstyle\sum x_i).$$

A threshold-gate computes a weighted sum of the inputs and tests whether the sum is greater than a threshold. In circuits, a threshold-gate with integer weights can be simulated by a majority-gate, because we can use parallel wires to simulate weights. Thus our results for circuits with majority-gates hold as well for circuits with threshold gates having bounded integral weights. The results for threshold circuits will not be stated explicitly.

3. Approximations

Rational approximations were developed by Newman [10]. Low-degree rational approximations are an important tool, which was applied to threshold circuit lower bounds by Paturi and Saks [12]. In order to prove upper bounds, we [6] observed that the small integral coefficients are also important.

Definition 1. A real-valued function $g(x_1, \ldots, x_n)$ *approximates* a function $f(x_1, \ldots, x_n)$ with error ϵ if for all $x_1, \ldots, x_n$ in the domain of f, $|g(x_1, \ldots, x_n) - f(x_1, \ldots, x_n)| \leq \epsilon$.

Lemma 2 ([6]). $\mathrm{MAJ}(x_1, \ldots x_n)$ *can be approximated with error ϵ by a rational function $g(x_1, \ldots, x_n)$ having degree $O(\log n \log(1/\epsilon))$ and integer coefficients bounded by $2^{O(\log^2 n \log(1/\epsilon))}$.*

Definition 3.

- The *norm* of a polynomial is the sum of the absolute values of its coefficients.
- The *norm* of a rational function is the norm of its numerator plus the norm of its denominator.

This definition of norm will be robust enough for our purposes because changing from the $\{0,1\}$ representation of Boolean values to the $\{-1,1\}$ representation changes the norm of a degree d rational function by a factor of only $O(2^d)$.

Note that the value of a polynomial over $\{0,1\}^k$ or over $\{-1,1\}^k$ is bounded by its norm. We need the following obvious corollary to the lemma of [6]:

Lemma 4. *Let n be any natural number.* $\mathrm{MAJ}(x_1, \ldots x_n)$ *can be approximated with error ϵ by a rational function $g(x_1, \ldots, x_n)$ having norm $2^{O(\log^2 n \log(1/\epsilon))}$.*

Proof: The number of monomials in a polynomial with n variables and degree d is $\binom{n+d}{d}$. Therefore the numerator and the denominator of the rational function in Lemma 2 contain $n^{O(\log n \log(1/\epsilon))}$ monomials, which is $2^{O(\log^2 n \log(1/\epsilon))}$. The coefficients are also $2^{O(\log^2 n \log(1/\epsilon))}$. □

4. Eliminating majority-gates

Let G be any set of gates that includes unbounded fanin AND-gates. Let TC_G(depth d, size s, majorities m) denote the class of circuits consisting of gates in G and majority-gates with depth d, size s, and m majority-gates. Let C_G(depth d, size s) denote TC_G(depth d, size s, majorities 0).

Theorem 5. *Suppose that f is in* TC_G(depth d, size s, majorities m). *Then f is in* TC_G(depth $d+2$, size $2^{m(O(\log s))^{2d+1}}$, majorities 1), *where the majority-gate is at the root.*

Taking $G = \{\mathrm{AND}, \mathrm{OR}, \mathrm{NOT}\}$ and $d = O(1)$, this exponentially improves the size bound of [6].

We extend the upper bounds of [14, 5].

Corollary 6. *Let* $G = \{\mathrm{AND}, \mathrm{OR}, \mathrm{NOT}\}$. *Every function in* TC_G(depth $O(1)$, size quasipolynomial, majorities polylog) *is computable by a probabilistic perceptron having polylog order.*

We unify some lower bounds of [4], [15, 9] and [13].

Corollary 7. *Let* $G = \{\mathrm{AND}, \mathrm{OR}, \mathrm{NOT}\}$. *If f agrees with parity on three-fourths of all inputs in* $\{0,1\}^n$ *then f is not in* TC_G(depth d, size $2^{n^{o(1/d^2)}}$, majorities $n^{o(1/d)}$).

Let $\mathrm{sgn}(x)$ denote the signum function: -1 if $x < 0$, 1 if $x > 0$, 0 if $x = 0$.

Proof of theorem: It will be convenient to assume that f's arguments are in $\{0,1\}$ and that f's result is in $\{-1,1\}$. Let $f : \{0,1\}^n \to \{-1,1\}$ belong to TC_G(depth d, size s, majorities m). We will show that $f = \mathrm{sgn}(p(f_1, \ldots, f_\ell))$ where p has norm $2^{m(O(\log s))^{2d+1}}$ and $f_1, \ldots, f_\ell$ belong to C_G(depth d, size s). (In fact if we view the original circuit as a DAG, $f_1, \ldots, f_\ell$ are some of its subgraphs.) That proves the theorem, because products can be computed by AND-gates when inputs are represented in $\{0,1\}$.

Let TC_G(depth d, size s, majorities m, level k) denote the class of TC_G(depth d, size s, majorities m) circuits with majority-gates on only levels 0 through k, where level 0 is the root. If f is computed by a TC_G(depth d, size s, majorities m, level k) circuit we will construct p and $f_1, \ldots, f_\ell$ such that $f = \mathrm{sgn}(p(f_1, \ldots, f_\ell))$, p is a polynomial whose norm is bounded by some function $N_p(k)$, and $f_1, \ldots, f_\ell$ belong to C_G(depth d, size s). We will find a recurrence for $N_p(k)$, and show that $N_p(k) = 2^{m(O(\log s))^{2k+1}}$. The theorem follows from that bound, taking $k = d$.

Consider a function f computed by a circuit with majority-gates only on levels 0 through k. We compute f by summing, over all sequences of possible outputs for the majority-gates on level k, (a) the value of f given those outputs (in (a) we use -1 for false and 1 for true) multiplied by (b) the AND of the corresponding majorities or their complement (in (b) we use 0 for false and 1 for true).

Each term in (a) is the sign of a polynomial p of C_G(depth d, size s) functions, where the norm of p is bounded by $N_p(k-1)$.

Suppose that there are t majority-gates at level k. Then $t \leq k$. The terms in (b) are products of exactly t factors, each of which is either a majority or its complement. Let $\epsilon = 1/(m(2^m N_p(k-1)+1))$. Each majority has at most s inputs, so it can be approximated within error ϵ by a rational function whose norm is $2^{O((\log^2 s)(m+\log N_p(k-1)))}$, by Lemma 4.

If a majority-gate is approximated within ϵ by the rational function r, then its complement is approximated within ϵ by the rational function $1 - r$, which has the same denominator as r. If r is as in the previous paragraph, then the norm of $1 - r$ is also $2^{O((\log^2 s)(m+\log N_p(k-1)))}$.

We approximate each term in (b) by the product of the rational functions that approximate the corresponding majorities or their complements. The error is at most $(1+\epsilon)^t - 1 \leq (1+\epsilon)^m - 1$. Furthermore the denominator is the same for each term because r and $1 - r$ have the same denominator.

The function f is then approximated by taking the sum of the 2^m terms (a) times (b). Since each term in (a) is bounded by $N_p(k-1)$, the error in approximating f is bounded by $2^m N_p(k-1)((1+\epsilon)^m - 1)$, which is less than 1 by Lemma 8 (proved below) with $N = N_p(k-1)$, so the approximation has the same sign as f.

Recall that all the rational functions used in the approximation have the same denominator. We obtain a polynomial that has the same sign as f by multiplying by the square of that common denominator. If N_r bounds the norm of the rational functions and $N_p(k-1)$ bounds the norm of the polynomials p used for (a) then $2^m N_p(k-1)N_r^2$ bounds the norm of the resulting polynomial, so

$$N_p(k) \leq 2^m N_p(k-1) 2^{O((\log^2 s)(m+\log N_p(k-1)))}.$$

Furthermore, we have $N_p(0) < s$. An easy induction shows that

$$N_p(k) = 2^{(m+\log s)(O(\log s))^{2k}} = 2^{m(O(\log s))^{2k+1}}.$$

□

Lemma 8. *Let* $\epsilon = 1/(m(2^m N + 1))$. *Then* $2^m N((1+\epsilon)^m - 1) < 1$.

Proof: For all real y, $1 + y \leq e^y$, with equality holding iff $y = 0$. By inverting that inequality and letting $y = -1/(1+x)$ where $x > 0$, we obtain

$$1 + 1/x > e^{\frac{1}{1+x}} = (e^{\frac{1}{m(1+x)}})^m > (1 + \frac{1}{m(x+1)})^m.$$

Letting $x = 2^m N$, we have

$$2^m N((1+\epsilon)^m - 1) < 2^m N(1 + 2^{-m} N^{-1} - 1) = 1,$$

as claimed. □

5. Eliminating Symmetric Gates

By applying some simple techniques developed by Papadimitriou and Zachos [11] in the study of #P, we will show how to reduce the number of symmetric gates in circuits to 1. Let SYMMC$_G$(depth d, size s, symmetrics m) denote the class of circuits consisting of gates in G and symmetric gates with depth d, size s, and m symmetric gates. (In this section G need not contain unbounded fanin AND-gates.)

Theorem 9. *Suppose that f is in* SYMMC_G(depth d, size s, symmetrics m). *Then f is in* SYMMC_G(depth $d+1$, size s^{2^m+2}, symmetrics 1), *where the symmetric gate is at the root.*

Proof: Consider any circuit C with m symmetric gates, $g_1, \ldots, g_m$ where the depth of g_{i-1} is at least as great as the depth of g_i for $i = 2, \ldots, m$. Trying all possible output values for $g_1, \ldots, g_{i-1}$ we find that g_i computes a symmetric function of one of 2^{i-1} subcircuits of C. Doing so for all i, we see that it is sufficient to evaluate $2^m - 1$ symmetric functions of subcircuits of C and to evaluate a subcircuit of C in the case that there is not a symmetric gate at the root. A fortiori, it suffices to evaluate 2^m symmetric functions of subcircuits of g. Let $M = 2^m$, and let $x_0, \ldots, x_M$ respectively denote the sum of the inputs to each of those symmetric functions. All of these numbers can be encoded into a single number, $X = \sum_{i=0}^{M} x_i B^i$, where B is any number larger than each of the x_i's; taking $B = s$ suffices. X may be computed by a single symmetric gate whose inputs are $B^{M+1} - 1$ subcircuits of C. The size of the resulting circuit is

$$s(B^{M+1} - 1) = s(s^{2^m+1} - 1) < s^{2^m+2}.$$

□

We extend the results of [16, 7] on ACC.

Corollary 10. *Let* $G = \{\mathrm{AND}, \mathrm{OR}, \mathrm{NOT}, \mathrm{MOD}_m\}$. *Then every function in* SYMMC_G(depth $O(1)$, size quasipolynomial, symmetrics $O(\log \log n)$) *is computable in depth two by a quasipolynomial size circuit with a symmetric gate at the root and polylog-fanin AND-gates at the leaves.*

6. Extensions

When we eliminate majority-gates, it is notable that unbounded-fanin AND-gates are not always required. The fanin of the AND-gates at the second level of the simulating circuits is bounded by the degree of the polynomial associated with the construction. Furthermore, we have noted that the subcircuits which feed into those AND-gates are subgraphs of the original circuit. Our size bounds are at most squared if we make separate copies of the inputs to those AND-gates. Therefore, all of our results about majority-gates go through for formulas as well as for circuits. Our results about symmetric gates go through directly for formulas.

Our results also apply to circuits and formulas with threshold gates having small weights. Some authors specify polynomial-bounded weights. A more general way of requiring small weights is to define the circuit size to be the number of wires plus the sum of all weights. Then our results go through for threshold gates in place of majority gates, because we may convert such threshold gates directly to rational functions with the same norm and degree bounds as in Lemma 4.

Acknowledgments. We thank Lance Fortnow and especially Nick Reingold for very helpful discussions; Roman Smolensky for suggesting that we try to reduce the number of symmetric gates in circuits; and Les Valiant for pointing out that our proofs are valid for formulas.

References

[1] E. Allender. A note on the power of threshold circuits. In *Proceedings of the 30th IEEE Symposium on Foundations of Computer Science*, pages 580–584, 1989.

[2] E. Allender and U. Hertrampf. On the power of uniform families of constant depth threshold circuits. In *Proceedings of the 15th International Symposium on Mathematical Foundations of Computer Science*, pages 158–164, Aug. 1990. Lecture Notes in Computer Science 452.

[3] A. Amir, R. Beigel, and W. I. Gasarch. Some connections between bounded query classes and non-uniform complexity. In *Proceedings of the 5th Annual Conference on Structure in Complexity Theory*, pages 232–243. IEEE Computer Society Press, July 1990.

[4] J. Aspnes, R. Beigel, M. Furst, and S. Rudich. The expressive power of voting polynomials. In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, pages 402–409, 1991.

[5] R. Beigel, N. Reingold, and D. Spielman. The perceptron strikes back. In *Proceedings of the 6th Annual Conference on Structure in Complexity Theory*, pages 286–291, 1991.

[6] R. Beigel, N. Reingold, and D. Spielman. PP is closed under intersection. In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, pages 1–9, 1991.

[7] R. Beigel and J. Tarui. On ACC. In *Proceedings of the 32nd IEEE Symposium on Foundations of Computer Science*, pages 783–792, 1991.

[8] L. Fortnow and N. Reingold. PP is closed under truth-table reductions. In *Proceedings of the 6th Annual Conference on Structure in Complexity Theory*, pages 13–15, 1991.

[9] J. Hastad. Almost optimal lower bounds for small depth circuits. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing*, pages 6–20, 1986.

[10] D. J. Newman. Rational approximation to $|x|$. *Michigan Mathematical Journal*, 11:11–14, 1964.

[11] C. H. Papadimitriou and S. K. Zachos. Two remarks on the complexity of counting. In *Proceedings of the 6th GI Conference on Theoretical Computer Science*, pages 269–276. Springer-Verlag, 1983. Volume 145 of *Lecture Notes in Computer Science.*

[12] R. Paturi and M. E. Saks. On threshold circuits for parity. In *Proceedings of the 31st IEEE Symposium on Foundations of Computer Science*, 1990. 397-404.

[13] K.-Y. Siu, V. Roychowdhury, and T. Kailath. Computing with almost optimal size threshold circuits. Manuscript. Available from Information Systems Laboratory, Stanford University, Stanford, CA 94305., 1990.

[14] J. Tarui. Randomized polynomials, threshold circuits, and the polynomial hierarchy. In *Proceedings of the 8th Annual Symposium on Theoretical Aspects of Computer Science.* Springer-Verlag, 1991.

[15] A. C. Yao. Separating the polynomial-time hierarchy by oracles. In *Proceedings of the 26th Annual IEEE Symposium on Foundations of Computer Science*, pages 1–10, 1985.

[16] A. C.-C. Yao. On ACC and threshold circuits. In *Proceedings of the 31st IEEE Symposium on Foundations of Computer Science*, pages 619–627, 1990.